

ЛОГИСТИЧЕСКАЯ РЕГРЕССИЯ

Сергей Николенко

СПбГУ — Санкт-Петербург

1 октября 2020 г.

Random facts:

- 1 октября 331 г. до н.э. серпоносные колесницы и даже слоны нисколько не помогли Дарию при Гавгамелах
- 1 октября 1908 г. Генри Форд представил Ford Model T (Tin Lizzie), который стоил всего 850 долларов; к 1927 году было выпущено 15 млн штук; а 1 октября 1960 г. с конвейера сошёл первый «Запорожец»
- 1 октября 1964 г. была запущена первая линия Синкансэн между Токио и Осакой: 515км поезд преодолевал за 2.5 часа
- 1 октября 1977 г. Пеле сыграл свой последний матч, в котором один тайм сыграл за нью-йоркский «Космос», а другой — за бразильский «Сантос»; всего Пеле за карьеру забил 1281 гол в 1363 играх
- 1 октября 1992 г. в России началась выдача приватизационных сертификатов — ваучеров

ЛОГИСТИЧЕСКАЯ РЕГРЕССИЯ

- Итак, мы рассмотрели логистический сигмоид:

$$p(C_1 | \mathbf{x}) = \frac{p(\mathbf{x} | C_1)p(C_1)}{p(\mathbf{x} | C_1)p(C_1) + p(\mathbf{x} | C_2)p(C_2)} = \frac{1}{1 + e^{-a}} = \sigma(a),$$

$$\text{где } a = \ln \frac{p(\mathbf{x} | C_1)p(C_1)}{p(\mathbf{x} | C_2)p(C_2)}, \quad \sigma(a) = \frac{1}{1 + e^{-a}}.$$

- Вывели из него LDA и QDA, обучили их методом максимального правдоподобия, а потом отвлеклись на naïve Bayes.

- Возвращаемся к задаче классификации.
- Два класса, и апостериорное распределение – логистический сигмоид на линейной функции:

$$p(C_1 | \phi) = y(\phi) = \sigma(\mathbf{w}^\top \phi), \quad p(C_2 | \phi) = 1 - p(C_1 | \phi).$$

- *Логистическая регрессия* – это когда мы напрямую оптимизируем \mathbf{w} .

- Для датасета $\{\phi_n, t_n\}$, $t_n \in \{0, 1\}$, $\phi_n = \phi(\mathbf{x}_n)$:

$$p(\mathbf{t} | \mathbf{w}) = \prod_{n=1}^N y_n^{t_n} (1 - y_n)^{1-t_n}, \quad y_n = p(C_1 | \phi_n).$$

- Ищем параметры максимального правдоподобия, минимизируя $-\ln p(\mathbf{t} | \mathbf{w})$:

$$E(\mathbf{w}) = -\ln p(\mathbf{t} | \mathbf{w}) = -\sum_{n=1}^N [t_n \ln y_n + (1 - t_n) \ln(1 - y_n)].$$

- Пользуясь тем, что $\sigma' = \sigma(1 - \sigma)$, берём градиент (похоже на перцептрон):

$$\nabla E(\mathbf{w}) = \sum_{n=1}^N (y_n - t_n) \phi_n.$$

- Если теперь сделать градиентный спуск, получим как раз разделяющую поверхность.
- Заметим, правда, что если данные действительно разделимы, то может получиться жуткий оверфиттинг: $\|\mathbf{w}\| \rightarrow \infty$, и сигмоид превращается в функцию Хевисайда. Надо регуляризовать.

- В логистической регрессии не получается замкнутого решения из-за сигмоида.
- Но функция $E(\mathbf{w})$ всё равно выпуклая, и можно воспользоваться методом Ньютона-Рапсона – на каждом шаге использовать локальную квадратичную аппроксимацию к функции ошибки:

$$\mathbf{w}^{\text{new}} = \mathbf{w}^{\text{old}} - H^{-1} \nabla E(\mathbf{w}),$$

где H (Hessian) – матрица вторых производных $E(\mathbf{w})$.

- Замечание: давайте применим Ньютона-Рапсона к обычной линейной регрессии с квадратической ошибкой:

$$\begin{aligned}\nabla E(\mathbf{w}) &= \sum_{n=1}^N (\mathbf{w}^\top \phi_n - t_n) \phi_n = \Phi^\top \Phi \mathbf{w} - \Phi^\top \mathbf{t}, \\ \nabla \nabla E(\mathbf{w}) &= \sum_{n=1}^N \phi_n \phi_n^\top = \Phi^\top \Phi,\end{aligned}$$

и шаг оптимизации будет

$$\begin{aligned}\mathbf{w}^{\text{new}} &= \mathbf{w}^{\text{old}} - (\Phi^\top \Phi)^{-1} [\Phi^\top \Phi \mathbf{w}^{\text{old}} - \Phi^\top \mathbf{t}] = \\ &= (\Phi^\top \Phi)^{-1} \Phi^\top \mathbf{t},\end{aligned}$$

т.е. мы за один шаг придём к решению.

- Для логистической регрессии:

$$\nabla E(\mathbf{w}) = \sum_{n=1}^N (y_n - t_n) \phi_n = \Phi^T (\mathbf{y} - \mathbf{t}),$$

$$H = \nabla \nabla E(\mathbf{w}) = \sum_{n=1}^N y_n (1 - y_n) \phi_n \phi_n^T = \Phi^T R \Phi$$

для диагональной матрицы R с $R_{nn} = y_n(1 - y_n)$.

- Формула шага оптимизации:

$$\mathbf{w}^{\text{new}} = \mathbf{w}^{\text{old}} - (\Phi^{\top} R \Phi)^{-1} \Phi^{\top} (\mathbf{y} - \mathbf{t}) = (\Phi^{\top} R \Phi)^{-1} \Phi^{\top} R \mathbf{z},$$

где $\mathbf{z} = \Phi \mathbf{w}^{\text{old}} - R^{-1} (\mathbf{y} - \mathbf{t})$.

- Получилось как бы решение взвешенной задачи минимизации квадратического отклонения с матрицей весов R .
- Отсюда название: iterative reweighted least squares (IRLS).

ЛОГИСТИЧЕСКАЯ РЕГРЕССИЯ

- Итак, мы рассмотрели логистический сигмоид:

$$p(C_1 | \mathbf{x}) = \frac{p(\mathbf{x} | C_1)p(C_1)}{p(\mathbf{x} | C_1)p(C_1) + p(\mathbf{x} | C_2)p(C_2)} = \frac{1}{1 + e^{-a}} = \sigma(a),$$

$$\text{где } a = \ln \frac{p(\mathbf{x} | C_1)p(C_1)}{p(\mathbf{x} | C_2)p(C_2)}, \quad \sigma(a) = \frac{1}{1 + e^{-a}}.$$

- Вывели из него LDA и QDA, обучили их методом максимального правдоподобия, а потом отвлеклись на naïve Bayes.

- Два класса, и апостериорное распределение – логистический сигмоид на линейной функции:

$$p(C_1 | \phi) = y(\phi) = \sigma(\mathbf{w}^T \phi), \quad p(C_2 | \phi) = 1 - p(C_1 | \phi).$$

- *Логистическая регрессия* – это когда мы напрямую оптимизируем \mathbf{w} .

- Для датасета $\{\phi_n, t_n\}$, $t_n \in \{0, 1\}$, $\phi_n = \phi(\mathbf{x}_n)$:

$$p(\mathbf{t} | \mathbf{w}) = \prod_{n=1}^N y_n^{t_n} (1 - y_n)^{1-t_n}, \quad y_n = p(C_1 | \phi_n).$$

- Ищем параметры максимального правдоподобия, минимизируя $-\ln p(\mathbf{t} | \mathbf{w})$:

$$E(\mathbf{w}) = -\ln p(\mathbf{t} | \mathbf{w}) = -\sum_{n=1}^N [t_n \ln y_n + (1 - t_n) \ln(1 - y_n)].$$

- Пользуясь тем, что $\sigma' = \sigma(1 - \sigma)$, берём градиент (похоже на перцептрон):

$$\nabla E(\mathbf{w}) = \sum_{n=1}^N (y_n - t_n) \phi_n.$$

- Если теперь сделать градиентный спуск, получим как раз разделяющую поверхность.
- А ещё лучше – IRLS, который мы обсуждали в прошлый раз.

- В случае нескольких классов

$$p(C_k | \phi) = y_k(\phi) = \frac{e^{a_k}}{\sum_j e^{a_j}} \text{ для } a_k = \mathbf{w}_k^\top \phi.$$

- Опять выпишем максимальное правдоподобие; во-первых,

$$\frac{\partial y_k}{\partial a_j} = y_k ([k = j] - y_j).$$

- Теперь запишем правдоподобие – для схемы кодирования 1-of- K будет целевой вектор \mathbf{t}_n и правдоподобие

$$p(T | \mathbf{w}_1, \dots, \mathbf{w}_K) = \prod_{n=1}^N \prod_{k=1}^K p(C_k | \phi_n)^{t_{nk}} = \prod_{n=1}^N \prod_{k=1}^K y_{nk}^{t_{nk}}$$

для $y_{nk} = y_k(\phi_n)$; берём логарифм:

$$E(\mathbf{w}_1, \dots, \mathbf{w}_K) = -\ln p(T | \mathbf{w}_1, \dots, \mathbf{w}_K) = -\sum_{n=1}^N \sum_{k=1}^K t_{nk} \ln y_{nk}, \text{ и}$$

$$\nabla_{\mathbf{w}_j} E(\mathbf{w}_1, \dots, \mathbf{w}_K) = -\sum_{n=1}^N (y_{nj} - t_{nj}) \phi_n.$$

- Оптимизировать опять можно по Ньютону-Рапсону; гессиан получится как

$$\nabla_{\mathbf{w}_k} \nabla_{\mathbf{w}_j} E(\mathbf{w}_1, \dots, \mathbf{w}_K) = - \sum_{n=1}^N y_{nk} ([k=j] - y_{nj}) \phi_n \phi_n^\top.$$

- А что если у нас другая форма сигмоида?
- Мы по-прежнему в той же постановке: два класса, $p(t = 1 | a) = f(a)$, $a = w^\top \phi$, f – функция активации.
- Давайте установим функцию активации с порогом θ : для каждого ϕ_n , вычисляем $a_n = w^\top \phi_n$, и

$$\begin{cases} t_n = 1, & \text{если } a_n \geq \theta, \\ t_n = 0, & \text{если } a_n < \theta. \end{cases}$$

- Если θ берётся по распределению $p(\theta)$, это соответствует

$$f(a) = \int_{-\infty}^a p(\theta) d\theta.$$

- Пусть, например, $p(\theta)$ – гауссиан с нулевым средним и единичной дисперсией. Тогда

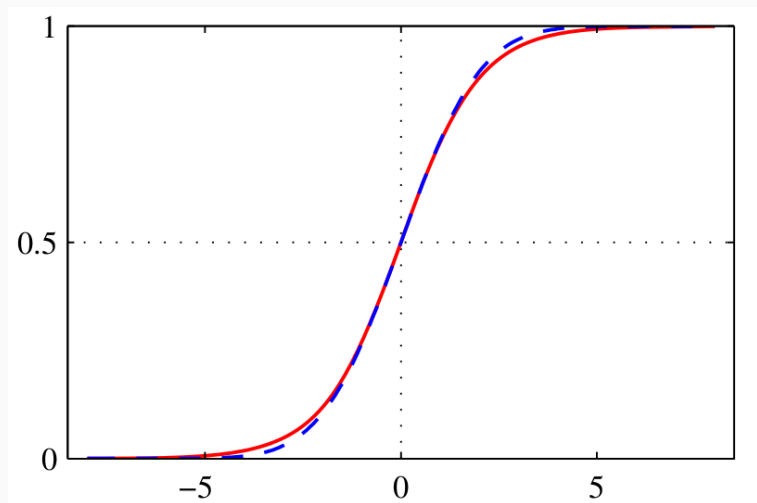
$$f(a) = \Phi(a) = \int_{-\infty}^a N(\theta | 0, 1) d\theta.$$

- Это называется *пробит-функцией* (probit); неэлементарная, но тесно связана с

$$\operatorname{erf}(a) = \frac{2}{\sqrt{\pi}} \int_0^a e^{-\frac{\theta^2}{2}} d\theta :$$

$$\Phi(a) = \frac{1}{2} \left[1 + \frac{1}{\sqrt{2}} \operatorname{erf}(a) \right].$$

- Пробит-регрессия – это модель с пробит-функцией активации.



ЛАПЛАСОВСКАЯ АППРОКСИМАЦИЯ
И
БАЙЕСОВСКАЯ
ЛОГИСТИЧЕСКАЯ РЕГРЕССИЯ

- Небольшое лирическое отступление: как приблизить сложное распределение простым?
- Например, как приблизить гауссианом возле максимума? (естественная задача)
- Рассмотрим пока распределение от одной непрерывной переменной $p(z) = \frac{1}{Z}f(z)$.

- Первый шаг: найдём максимум z_0 .
- Второй шаг: разложим в ряд Тейлора

$$\ln f(z) \approx \ln f(z_0) - \frac{1}{2}A(z - z_0)^2, \text{ где } A = -\frac{d^2}{dz^2} \ln f(z) \Big|_{z=z_0}.$$

- Третий шаг: приблизим

$$f(z) \approx f(z_0)e^{-\frac{A}{2}(z-z_0)^2},$$

и после нормализации это будет как раз гауссиан.

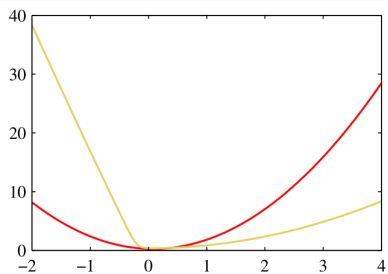
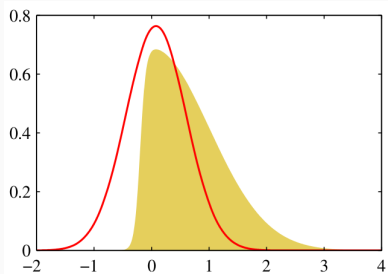
- Это можно обобщить на многомерное распределение $p(\mathbf{z}) = \frac{1}{Z} f(\mathbf{z})$:

$$f(\mathbf{z}) \approx f(\mathbf{z}_0) e^{-\frac{1}{2}(\mathbf{z}-\mathbf{z}_0)^\top \mathbf{A}(\mathbf{z}-\mathbf{z}_0)},$$

$$\text{где } \mathbf{A} = -\nabla \nabla \ln f(\mathbf{z}) \big|_{z=z_0}.$$

Упражнение. Какая здесь будет нормировочная константа?

ЛАПЛАСОВСКАЯ АППРОКСИМАЦИЯ



СРАВНЕНИЕ МОДЕЛЕЙ ПО ЛАПЛАСУ

- Вооружившись лапласовской аппроксимацией, давайте применим её сначала к выбору моделей.
- Напомним: чтобы сравнить модели из множества $\{M_i\}_{i=1}^L$, по тестовому набору D оценим апостериорное распределение

$$p(M_i | D) \propto p(M_i)p(D | M_i).$$

- Если модель определена параметрически, то $p(D | M_i) = \int p(D | \theta, M_i)p(\theta | M_i)d\theta$.
- Это вероятность сгенерировать D , если выбирать параметры модели по её априорному распределению; знаменатель из теоремы Байеса:

$$p(\theta | M_i, D) = \frac{p(D | \theta, M_i)p(\theta | M_i)}{p(D | M_i)}.$$

СРАВНЕНИЕ МОДЕЛЕЙ ПО ЛАПЛАСУ

- Мы раньше приближали фактически кусочно-постоянной функцией.
- Теперь давайте гауссианом приблизим; возьмём интеграл:

$$Z = \int f(\mathbf{z}) d\mathbf{z} \approx \int f(\mathbf{z}_0) e^{-\frac{1}{2}(\mathbf{z}-\mathbf{z}_0)^\top \mathbf{A}(\mathbf{z}-\mathbf{z}_0)} d\mathbf{z} = f(\mathbf{z}_0) \frac{(2\pi)^{M/2}}{|\mathbf{A}|^{1/2}}.$$

- А у нас $Z = p(D)$, $f(\theta) = p(D | \theta)p(\theta)$.

- Получаем

$$\ln p(D) \approx \ln p(D | \theta_{\text{MAP}}) + \ln P(\theta_{\text{MAP}}) + \frac{M}{2} \ln(2\pi) - \frac{1}{2} \ln |\mathbf{A}|.$$

- $\ln P(\theta_{\text{MAP}}) + \frac{M}{2} \ln(2\pi) - \frac{1}{2} \ln |\mathbf{A}|$ – фактор Оккама.
- $\mathbf{A} = -\nabla\nabla \ln p(D | \theta_{\text{MAP}})p(\theta_{\text{MAP}}) = -\nabla\nabla \ln p(\theta_{\text{MAP}} | D)$.

- Получаем

$$\ln p(D) \approx \ln p(D | \theta_{\text{MAP}}) + \ln P(\theta_{\text{MAP}}) + \frac{M}{2} \ln(2\pi) - \frac{1}{2} \ln |\mathbf{A}|.$$

- Если гауссовское априорное распределение $p(\theta)$ достаточно широкое, и \mathbf{A} полного ранга, то можно грубо приблизить (докажите это!)

$$\ln p(D) \approx \ln p(D | \theta_{\text{MAP}}) - \frac{1}{2} M \ln N,$$

где M – число параметров, N – число точек в D , а аддитивные константы мы опустили.

- Это *байесовский информационный критерий* (Bayesian information criterion, BIC), он же *критерий Шварца* (Schwarz criterion).

- Теперь давайте обработаем логистическую регрессию по-байесовски.
- Логистическую регрессию так просто не выпишешь, как линейную – точного ответа из произведения логистических сигмоидов не получается.
- Будем приближать по Лапласу.

- Априорное распределение выберем гауссовским:

$$p(\mathbf{w}) = N(\mathbf{w} \mid \mu_0, \Sigma_0).$$

- Тогда апостериорное будет

$$\begin{aligned} p(\mathbf{w} \mid \mathbf{t}) &\propto p(\mathbf{w})p(\mathbf{t} \mid \mathbf{w}), \text{ и} \\ \ln p(\mathbf{w} \mid \mathbf{t}) &= -\frac{1}{2} (\mathbf{w} - \mu_0)^\top \Sigma_0^{-1} (\mathbf{w} - \mu_0) \\ &\quad + \sum_{n=1}^N [t_n \ln y_n + (1 - t_n) \ln(1 - y_n)] + \text{const}, \\ \text{где } y_n &= \sigma(\mathbf{w}^\top \phi_n). \end{aligned}$$

- Чтобы приблизить, сначала находим максимум \mathbf{w}_{MAP} , а потом матрица ковариаций – это матрица вторых производных

$$\Sigma_N = -\nabla\nabla \ln p(\mathbf{w} | \mathbf{t}) = \Sigma_0^{-1} + \sum_{n=1}^N y_n(1 - y_n)\phi_n\phi_n^\top.$$

- Наше приближение – это

$$q(\mathbf{w}) = N(\mathbf{w} | \mathbf{w}_{\text{MAP}}, \Sigma_N).$$

- Теперь можно описать байесовское предсказание:

$$p(C_1 | \phi, \mathbf{t}) = \int p(C_1 | \phi, \mathbf{w})p(\mathbf{w} | \mathbf{t})d\mathbf{w} \approx \int \sigma(\mathbf{w}^\top \phi)q(\mathbf{w})d\mathbf{w}.$$

- Заметим, что $\sigma(\mathbf{w}^\top \phi)$ зависит от \mathbf{w} только через его проекцию на ϕ .
- Обозначим $a = \mathbf{w}^\top \phi$:

$$\sigma(\mathbf{w}^\top \phi) = \int \delta(a - \mathbf{w}^\top \phi)\sigma(a)da.$$

- $\sigma(\mathbf{w}^\top \phi) = \int \delta(a - \mathbf{w}^\top \phi) \sigma(a) da$, а значит,

$$\int \sigma(\mathbf{w}^\top \phi) q(\mathbf{w}) d\mathbf{w} = \int \sigma(a) p(a) da,$$

$$\text{где } p(a) = \int \delta(a - \mathbf{w}^\top \phi) q(\mathbf{w}) d\mathbf{w}.$$

- $p(a)$ – это маргинализация гауссиана $q(\mathbf{w})$, где мы интегрируем по всему, что ортогонально ϕ .

- $p(a)$ – это маргинализация гауссиана $q(\mathbf{w})$, где мы интегрируем по всему, что ортогонально ϕ .
- Значит, $p(a)$ – тоже гауссиан; найдём его моменты:

$$\begin{aligned}\mu_a = \mathbb{E}[a] &= \int a p(a) da = \int q(\mathbf{w}) \mathbf{w}^\top \phi d\mathbf{w} = \mathbf{w}_{\text{MAP}}^\top \phi, \\ \sigma_a^2 &= \int (a^2 - \mathbb{E}[a])^2 p(a) da = \\ &= \int q(\mathbf{w}) [(\mathbf{w}^\top \phi)^2 - (\mu_N^\top \phi)^2]^2 d\mathbf{w} = \phi^\top \Sigma_N \phi.\end{aligned}$$

- Итого получили, что

$$p(C_1 | \mathbf{t}) = \int \sigma(a) p(a) da = \int \sigma(a) N(a | \mu_a, \sigma_a^2) da.$$

- $p(C_1 | \mathbf{t}) = \int \sigma(a)N(a | \mu_a, \sigma_a^2)da.$
- Этот интеграл так просто не взять, потому что сигмоид сложный, но можно приблизить, если приблизить $\sigma(a)$ через пробит: $\sigma(a) \approx \Phi(\lambda a)$ для $\lambda = \sqrt{\pi/8}$.

Упражнение. Докажите, что для $\lambda = \sqrt{\pi/8}$ у σ и Φ одинаковый наклон в нуле.

- А если мы перейдём к пробит-функции, то её свёртка с гауссианом будет просто другим пробитом:

$$\int \Phi(\lambda a) N(a \mid \mu, \sigma^2) da = \Phi\left(\frac{\mu}{\sqrt{\frac{1}{\lambda^2} + \sigma^2}}\right).$$

Упражнение. Докажите это.

- В итоге получается аппроксимация

$$\int \sigma(a) N(a | \mu, \sigma^2) da \approx \sigma(\kappa(\sigma^2)\mu),$$

$$\text{где } \kappa(\sigma^2) = \frac{1}{\sqrt{1 + \frac{\pi}{8}\sigma^2}}.$$

- И теперь, собирая всё вместе, мы получили распределение предсказаний:

$$p(C_1 | \phi, \mathbf{t}) = \sigma(\kappa(\sigma_a^2)\mu_a), \text{ где}$$

$$\mu_a = \mathbf{w}_{\text{MAP}}^\top \phi,$$

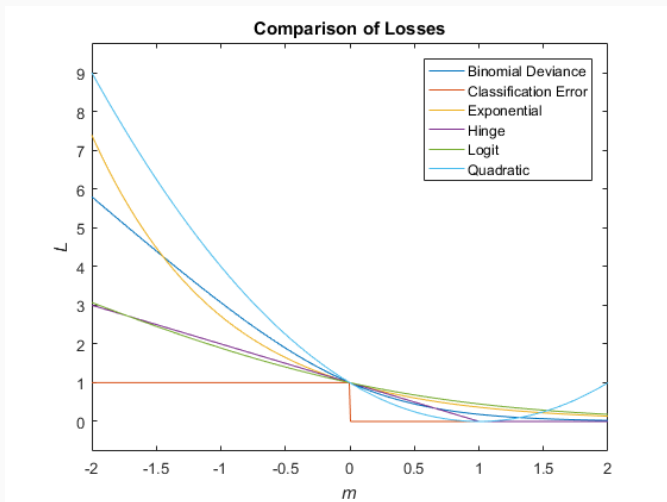
$$\sigma_a^2 = \phi^\top \Sigma_N \phi,$$

$$\kappa(\sigma^2) = \frac{1}{\sqrt{1 + \frac{\pi}{8}\sigma^2}}.$$

- Кстати, разделяющая поверхность $p(C_1 | \phi, \mathbf{t}) = \frac{1}{2}$ задаётся уравнением $\mu_a = 0$, и тут нет никакой разницы с просто использованием \mathbf{w}_{MAP} . Разница будет только для более сложных критериев.

- И напоследок немножко другой взгляд: разные методы классификации отличаются друг от друга тем, какую функцию ошибки они оптимизируют.
- У классификации проблема с «правильной» функцией ошибки, то есть ошибкой собственно классификации:
 - она и не везде дифференцируема,
 - и производная её никому не нужна.
- Давайте посмотрим на разные функции потерь (loss functions); мы уже несколько видели, но ещё немало осталось.

ФУНКЦИИ ПОТЕРЬ В КЛАССИФИКАЦИИ

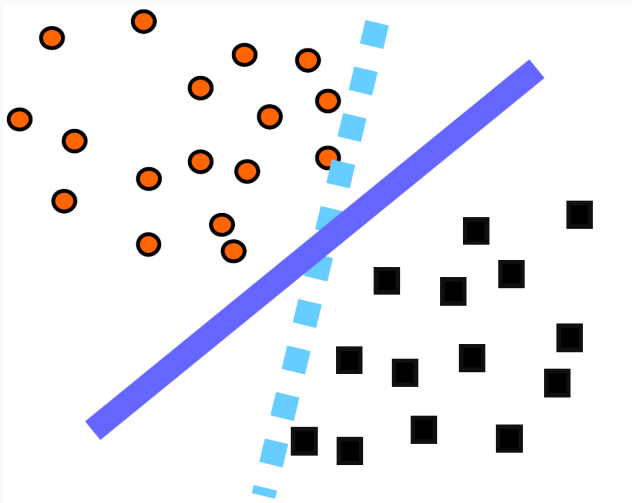


SVM и ЗАДАЧА ЛИНЕЙНОЙ КЛАССИФИКАЦИИ

ПОСТАНОВКА ЗАДАЧИ

- Метод опорных векторов решает задачу классификации.
- Каждый элемент данных — точка в n -мерном пространстве \mathbb{R}^n .
- Формально: есть точки $x_i, i = 1..m$, у точек есть метки $y_i = \pm 1$.
- Мы интересуемся: можно ли разделить данные $(n - 1)$ -мерной гиперплоскостью, а также хотим найти эту гиперплоскость.
- Это всё?

- Нет, ещё хочется научиться разделять этой гиперплоскостью *как можно лучше*.
- То есть желательно, чтобы два разделённых класса лежали как можно дальше от гиперплоскости.
- Практическое соображение: тогда от небольших возмущений в гиперплоскости ничего не испортится.

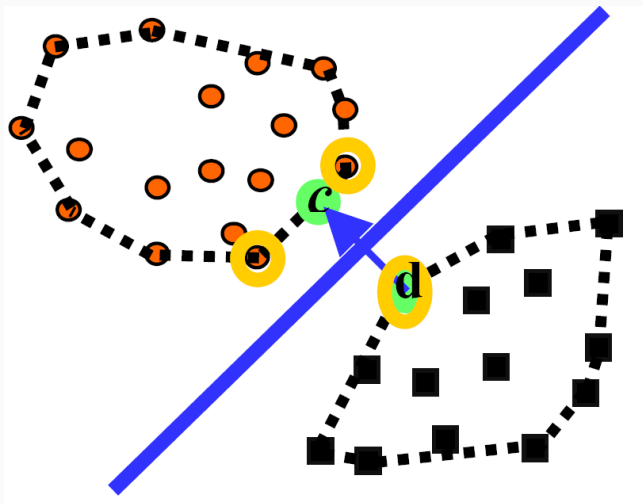


- Один подход: найти две ближайшие точки в выпуклых оболочках данных, а затем провести разделяющую гиперплоскость через середину отрезка.
- Формально это превращается в задачу квадратичной оптимизации:

$$\min_{\alpha} \left\{ \|c - d\|^2, \text{ где } c = \sum_{y_i=1} \alpha_i x_i, d = \sum_{y_i=-1} \alpha_i x_i \right\}$$

при условии $\sum_{y_i=1} \alpha_i = \sum_{y_i=-1} \alpha_i = 1, \alpha_i \geq 0.$

- Эту задачу можно решать общими оптимизационными алгоритмами.



Спасибо за внимание!